

### **maxSpan – lapsoMax**

Considere las apariciones más a la izquierda y más a la derecha de algún valor en el arreglo. Diremos que el “lapso” (span) es el número de elementos que hay entre dos elementos cualesquiera, incluyendo ambos extremos. Un valor sólo tiene lapso de 1. Retorne el lapso más grande encontrado en el arreglo (la eficiencia no es una prioridad).

### **fix34 – fijo34**

Retorne un arreglo que contiene exactamente los mismos números que el arreglo original, pero reordenados de manera que todo 3 está seguido inmediatamente por un 4. No mueva el 3, pero cualquier otro número puede moverse. El arreglo contiene el mismo número de 3s y 4s, cada 3 tiene un número después de este que no es ni 3 ni 4, y 3 aparece en el arreglo antes que cualquier 4.

### **fix45 – fijo45**

(Esta es una versión un poco más difícil del problema fijo34 (fix34)). Retorne un arreglo que contiene exactamente los mismos números que el arreglo original, pero reordenados de manera que cada 4 está seguido inmediatamente de un 5. No mueva el 4 pero cualquier otro número puede moverse. El arreglo contiene la misma cantidad de 4s y 5s y todo 4 tiene un número luego de sí que es diferente de 4. En esta versión, los 5s pueden aparecer en cualquier lugar del arreglo original.

### **canBalance – puedeBalancear**

Dado un arreglo no vacío de enteros, retorne verdadero si hay un lugar en el arreglo en el cual la suma de los valores a la derecha de este, incluyéndolo ([0, k]) es igual a la suma de los valores a la izquierda de este ([k+1, longitud-1]).

### **linearIn – internoLineal**

Dados dos arreglos de enteros ordenados en orden no-descendente, llamados externo (outer) e interno (inner), retorne verdadero si todos los números del arreglo interno están contenidos en el arreglo externo. La mejor solución realiza únicamente una pasada “lineal” de ambos arreglos, tomando ventaja del hecho de ambos arreglos están ordenados bajo el mismo criterio.

### **squareUp – cuadrar**

Dado un entero  $n \geq 0$ , cree un arreglo de longitud  $n \times n$  con el siguiente patrón, mostrado aquí para  $n=3$ : {0, 0, 1, 0, 2, 1, 3, 2, 1} (los espacios fueron añadidos para poder diferenciar los 3 grupos con más facilidad).

### **seriesUp – seriar**

Dado un entero  $n \geq 0$ , cree un arreglo con el patrón {1, 1, 2, 1, 2, 3, ... 1, 2, 3, ... n} (los espacios fueron añadidos para poder diferenciar los grupos con más facilidad). Note que la longitud del arreglo será  $1+2+3+\dots+n$ , lo cual se sabe que suma exactamente  $n \times (n+1)/2$ .

### **maxMirror – espejoMax**

Diremos que una sección “espejo” en un arreglo es un grupo de elementos continuos tales que en algún lugar en el arreglo, el mismo grupo aparece en orden inverso. Por ejemplo, la sección espejo más grande en el arreglo {1, 2, 3, 8, 9, 3, 2, 1} es de longitud 3 (la parte {1, 2, 3}). Retorne el tamaño de la sección espejo más grande encontrada en el arreglo dado.

### **countClumps – contarGrupos**

Diremos que un “grupo” (clump) en un arreglo es una serie de dos o más elementos adyacentes del mismo valor. Retorne el número de grupos en el arreglo dado.